Claims

[c1] 1. A method for controlling interprocess communication, the method comprising:
defining rules indicating which system services a given application can invoke;
trapping an attempt by a particular application to invoke a particular system service;
identifying the particular application that is attempting to invoke the particular system service; and
based on identity of the particular application and on the rules indicating which system services a given application can invoke, blocking the attempt when the rules indicate that the particular application cannot invoke the particular system service.

[c2] 2. The method of claim 1, wherein said trapping step includes intercepting operating system calls for invoking the particular system service.

[c3] 3. The method of claim 1, wherein said trapping step includes intercepting local procedure calls for invoking the particular system service.

[c4] 4. The method of claim 1, wherein said trapping step in-

cludes intercepting an attempt to open a communication channel to the particular system service.

[c5] 5. The method of claim 1, wherein said trapping step includes rerouting an attempt to invoke the particular system service from a system dispatch table to an interprocess communication controller for determining whether to block the attempt based on the rules.

[c6] 6. The method of claim 5, wherein said step of rerouting attempts to invoke the particular system service from a dispatch table to the interprocess communication controller includes replacing an original destination address in the system dispatch table with an address of the interprocess communication controller.

[c7] 7. The method of claim 6, further comprising the steps of:
retaining the original destination address; and
using the original destination address for invoking the particular system service if the interprocess communication controller determines not to block the attempt.

[c8] 8. The method of claim 1, wherein the rules specifying which system services a given application can invoke are established based on user input.

[c9] 9. The method of claim 1, wherein the step of blocking

the attempt is based upon consulting a rules engine for determining whether the particular application can invoke the particular system service.

[c10] 10. The method of claim 1, wherein the step of blocking the attempt includes obtaining user input as to whether the particular application can invoke the particular system service.

[c11] 11. The method of claim 10, wherein said step of obtaining user input as to whether the particular application can invoke the particular system service includes the substeps of:
providing information to the user about the particular application that is attempting to invoke the particular system service; and
receiving user input as to whether the particular application should be blocked from invoking the particular system service.

[c12] 12. A computer-readable medium having computer-executable instructions for performing the method of claim 1.

[c13] 13. A downloadable set of computer-executable instructions for performing the method of claim 1.

[c14] 14. In a computer system, a method for regulating com-

munications between processes, the method comprising:
defining a policy specifying whether one process may communicate with another process;
intercepting an attempt by a first process to communicate with a second process;
identifying the first process that is attempting to communicate with the second process;
identifying the second process;
based on said policy, determining whether the first process may communicate with the second process; and
allowing the first process to communicate with the second process if said policy indicates that the first process may communicate with the second process.

[c15] 15. The method of claim 14, wherein the first process comprises an instance of an application program.

[c16] 16. The method of claim 14, wherein the second process comprises a system service.

[c17] 17. The method of claim 14, wherein said intercepting step includes intercepting operating system calls made by the first process to attempt to communicate with the second process.

[c18] 18. The method of claim 14, wherein said intercepting step includes detecting local procedure calls.

[c19] 19. The method of claim 14, wherein said intercepting step includes detecting an attempt by the first process to open a communication channel to the second process.

[c20] 20. The method of claim 14, wherein said intercepting step includes rerouting attempts by the first process to communicate with the second process from a system dispatch table to an interprocess communication controller.

[c21] 21. The method of claim 14, wherein said step of identifying the second process includes evaluating parameters of the attempt made by the first process to communicate with the second process.

[c22] 22. The method of claim 14, wherein said policy specifies particular processes to be protected from communications made by other processes.

[c23] 23. The method of claim 14, further comprising:
providing for a process to be registered in order to be protected from communications made by other processes; and
determining whether to allow the first process to communicate with the second process based, at least in part, upon determining whether the second process is registered.

[c24] 24. The method of claim 23, wherein said determining step is based, at least in part, on the type of communication the first process is attempting with the second process.

[c25] 25. A method for controlling interprocess communications from one application to another, the method comprising:
registering a first application to be protected from other applications;
detecting an attempt to access the first application using interprocess communication;
identifying a second application that is attempting to access the first application using interprocess communication; and
rerouting the attempt to access the first application through an interprocess communication controller that determines whether to allow the attempt, based on rules indicating whether the second application may access the first application using interprocess communication.

[c26] 26. The method of claim 25, wherein said registering step includes supplying rules specifying particular communications from which the first application is to be protected.

[c27] 27. The method of claim 26, wherein the interprocess communication controller determines whether to allow the attempt based, at least in part, upon the rules specifying particular communications from which the first application is to be protected.

[c28] 28. The method of claim 25, wherein said detecting step includes intercepting operating system calls for accessing the first application.

[c29] 29. The method of claim 25, wherein said detecting step includes detecting a graphical device interface (GDI) message sent to the first application.

[c30] 30. The method of claim 29, wherein said identifying step includes evaluating parameters of the message sent to the first application.

[c31] 31. The method of claim 25, wherein said detecting step includes detecting an attempt to send keystoke data to a window of the first application.

[c32] 32. The method of claim 25, wherein said detecting step includes detecting an attempt to send mouse movement data to a window of the first application.

[c33] 33. The method of claim 25, wherein said rerouting step includes rerouting the attempt to access the first appli-

cation from a system dispatch table to the interprocess communication controller.

[c34] 34. The method of claim 25, wherein said rules indicating whether the second application may access the first application includes rules indicating particular types of communications which are allowed.

[c35] 35. The method of claim 25, further comprising:
if the interprocess communication controller allows the attempt to access the first application, routing the attempt to the first application.

[c36] 36. A system for regulating interprocess communication between applications, the system comprising:
a policy specifying applications that are permitted to communicate with a first application using interprocess communication;
a module for detecting a second application attempting to communicate with the first application using interprocess communication; and
an interprocess communication controller for identifying the second application attempting to communicate with the first application and determining whether to permit the communication based upon the identification of the second application and the policy specifying applications permitted to communicate with the first application.

[c37] 37. The system of claim 36, wherein said policy includes rules indicating particular types of communications which are permitted.

[c38] 38. The system of claim 36, further comprising:
a rules engine for specifying applications that are permitted to communicate with the first application using interprocess communication.

[c39] 39. The system of claim 36, further comprising:
a registration module for establishing said policy.

[c40] 40. The system of claim 39, wherein said registration module provides for identifying applications to be governed by said policy.

[c41] 41. The system of claim 36, wherein said module for detecting a second application detects an operating system call to open a communication channel to the first application.

[c42] 42. The system of claim 36, wherein said module for detecting a second application detects a graphical device interface (GDI) message sent to the first application.

[c43] 43. The system of claim 36, wherein said module for detecting a second application detects a local procedure call attempting to access the first application.

[c44] 44. The system of claim 36, wherein said module for detecting a second application redirects attempts to communicate with the first application to the interprocess communication controller.

[c45] 45. The system of claim 36, wherein said module for detecting a second application reroutes the attempt to communicate with the first application from a dispatch table to the interprocess communication controller.

[c46] 46. The system of claim 36, wherein said interprocess communication controller determines whether to permit the communication based, at least in part, upon evaluating parameters of the attempt made by the second application to communicate with the first application.

[c47] 47. The system of claim 36, wherein said interprocess communication controller determines whether to permit the communication based upon obtaining user input as to whether to permit the second application to communicate with the first application.